International Journal of Advances in Intelligent Informatics
Vol. 5, No. 2, July 2019, pp. 101-112
ISSN 2442-6571

Reversible data hiding method by extending reduced difference expansion

Zainal Syahlan [a,b,1,*], Tohari Ahmad [a,2]

[a] Department of Informatics, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia
[b] Study Program of Informatics, Sekolah Tinggi Teknologi Angkatan Laut (STTAL), Surabaya, Indonesia
[1] zsyahlan@gmail.com; [2] tohari@if.its.ac.id
* corresponding author

ARTICLE INFO

Article history
Received April 16, 2019
Revised June 14, 2019
Accepted June 14, 2019
Available online July 25, 2019

Keywords
Steganography
Difference expansion
Reduced difference expansion
Hiding data
Secret data

ABSTRACT

To keep hiding secret data in multimedia files, such as video, audio, and image considers essential for information security. Image, for instance, as the media aids data insertion securely. The use of insertion technique must ensure a reliable process on retaining data quality and capacity. However, a trade-off between the resulted image quality and the embedded payload capacity after the embedding process often occurs. Therefore, this research aims at extending the existing method of integrating confidential messages using the Reduced Difference Expansion (RDE), transform into a medical image by changing the base point, block size, and recalculating of difference. The results display that the proposed method enhances the quality of the stego image and capacity of the hidden message.

1. Introduction

Vast globalization has brought significant influence on human life, mainly information and communication technology. Consequently, the data dispatching process becomes quickly through various kinds of media. However, a problem occurs, namely, the security problem of the data being sent rapid development in the process of sending data [1]. To save the data securely, the necessity of specific methods proposed systematically throughout this paper.

In this case, the security process for data to be sent mainly operated by encrypting a file. Data encoding takes cryptography and steganography as the main functional types to secure data in a file. The encoded and encrypted data were converted into a particular symbol in a purpose to keep secure. Other words, only certain people can find out the contents of the data [2],[3]. A different way to save the data securely, steganography operates scientifically hiding data with specific techniques [4],[5]. As the most effective way to reduce suspicion from other parties, steganography facilitates keeping the data safe during transmission. Steganography techniques used for information in multimedia files such as text, image, audio, and video [6], [7] illustrates the need for an essential role in the world of multimedia, especially in terms of data security [8],[9].

In a reversible data hiding scheme, the cover media can be taken entirely from the stego media after extracting the original message [10], [11]. Generally, two approaches of digital image stenography, namely, spatial domains and transformation. The former, spatial domain, none of the transformations occurs before hiding the secret message on the cover image. Studies of spatial domain approaches for protecting confidential data have conducted [12]-[14]. In transforming embedding a secret message, the image is changed from the spatial domain to frequency using several transformation schemes such as discrete cosine transform (DCT), discrete wavelet transformation (DWT), curvelet transformation, etc.

Then the secret message is embedded in the transformation coefficient. Above shows its superiority over the spatial domain [15], [16]. Research has also been carried out by references [17]-[19] wherein the research carried out by applying the secret data hiding approach in audio. First, the audio is processed; next, the hidden data stored in the least significant bit (LSB). In the previous research [20], medical data protected in an audio file with a particular data hiding scheme.

Different Expansion (DE) method [21] proposed in 2003, was one of the popular reversible data embedding algorithms. The DE method works not only by inserting data bits on the difference between pixels but also maintaining the average value. In that proposed method, secret data kept inside the expansion difference between two pixels. The research in Alattar [22] and Alattar [23] developed the DE method by replacing the number of pixels in blocks from two to three to increase the insertion capacity; while in Thodi and Rodríguez [24], the research combines the difference expansion with the prediction-error development to improve the image quality. In 2007, Liu et al. [25] proposed a Reduced Difference Expansion (RDE) method, which was also intended to improve the image quality. It reduces the difference in pixels before the data insertion process. In 2009, Yi et al. [26] modified the RDE method by embedding data and extracting data, where the stored data obtained correctly. Also, the original image stays fully recovered without distortion. In 2013, Ahmad et al. [27] and Kurniawan et al. [28] proposed an algorithm that could improve the quality of stego images, especially in the case of PSNR, and maintain the secret message. Furthermore, Al Huti et al. [29] used the pixel value of 4×4 blocks and could hide confidential data 15 bits per block; and reduced the number of the location maps.

Data hiding has been spreading in various fields, including in the health environment. For example, it is used to insert secret patient data into the media before being sent. Nevertheless, this storing classified data information on media such as images is a challenging problem. In general, medical images cannot tolerate any noise. The noise causes a diagnostic error. In this case, the method used must be reversible, so that it can return the image to its original state, where RDE is among them. In this case, various methods have been developed to detect information during transmission [30], [31].

RDE generally returns the stego image revert to the original cover image, but there are still problems with message embedding, such as the similarity between the unique and the stego image, and the capacity of the data dispensed back to the original image. In addition to the reversibility factor, overcoming these issues is still a challenge. In this study, we propose an RDE-based data hiding method for hiding data based on expansion points. In this method, we find an appropriate based point, the size of block pixels, which leads to getting another calculation of the difference between pixels. We use a reduction of pixel values to improve functional similarity; the goal is to have high quality in the image. In this method, we use 2×2 pixel per block.

This paper consists of four sections. In Section 1, we explain the research background. While in Section 2, we describe the proposed method, and in Section 3, we analyze and discuss the results. Section 4 presents the conclusion.

2. Method

The proposed method aims to improve data hiding techniques by utilizing pixel blocks and reducing the expansion differences calculated between pairs of pixels in the block. The proposed method is designed to improve existing technical abilities, especially those presented by previous researchers [29]. Therefore, the contribution of this research is to improve the performance of embedding the secret data on an image by extending the RDE method. It enhances the reduction scheme and varies the pixel block size that can hide 3 bits per block of size 2×2 as in Fig. 1.

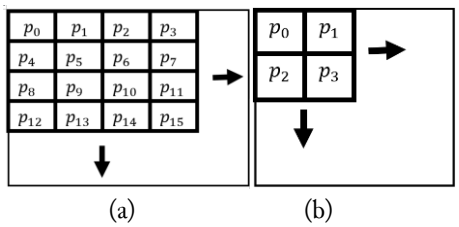

Fig. 1. The difference in pixel blocks (a) the method of Al Huti et al. [29] (b) the proposed method

2.1. Data Embedding Stage

The following description illustrates three stages of embedding process:

Step 1: Divide the image into blocks of 2×2, so each block has 4 pixels. In Fig. 1(b) one-pixel block is represented by p_0, p_1, p_2 and p_3. Then, before calculating the difference between pairs of pixels, all pixels in each block are stored as vectors. If p_0, p_1, p_2 and p_3 are pixels in the first block, vectors defined as v_{vec} = (p_0, p_1, p_2 and p_3). This method is the same as what was proposed in Al Huti et al. [29], where pixel blocks are categorized firstly into three groups, namely expandable, changeable, and non-changeable. Secret data is only embedded in the first and second groups to avoid overflow and underflow problems. Data on scalable blocks are hidden using (1), while (2) is used for changeable blocks. Then, non-changeable blocks, which can cause the underflow or overflow problem, is ignored during the embedding process. Here, v_n and v'_n are the difference before and after being embedded with the message S_n, respectively. Unlike the previous methods, this proposed approach explores the third pixel of each block to be the base point, depicted in Fig. 2. The result shown that the base point of the block is p_2.

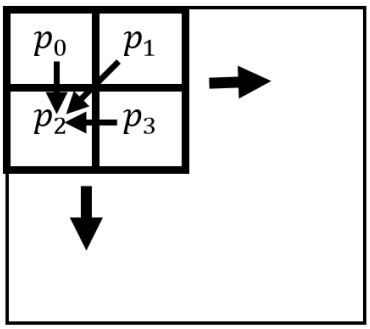

Fig. 2. Proposed base point pixel of a block (p_2)

$$\left.\begin{aligned} v'_0 &= 2 \times v_0 + S_0 \\ v'_1 &= 2 \times v_1 + S_1 \\ v'_3 &= 2 \times v_3 + S_3 \end{aligned}\right\} \quad (1)$$

$$\left.\begin{aligned} v'_0 &= 2 \times \left\lfloor \frac{v_0}{2} \right\rfloor + S_0 \\ v'_1 &= 2 \times \left\lfloor \frac{v_1}{2} \right\rfloor + S_1 \\ v'_3 &= 2 \times \left\lfloor \frac{v_3}{2} \right\rfloor + S_3 \end{aligned}\right\} \quad (2)$$

Step 2: Each block is defined, and the difference between pairs of pixels is calculated using (3). However, it is different from the previous method that p_2 Is not used to hide secret data here. That is, p_2 is not used ($p_2 = 0$) because the third pixel in each block took as the base point. Furthermore, to prevent the cover image from getting worse, the secret data are not embedded in the third pixel of each block (p_2). We propose an integer transformation (3) to calculate the difference using fixed pixel values as the base point seen as illuminates in Fig. 3.

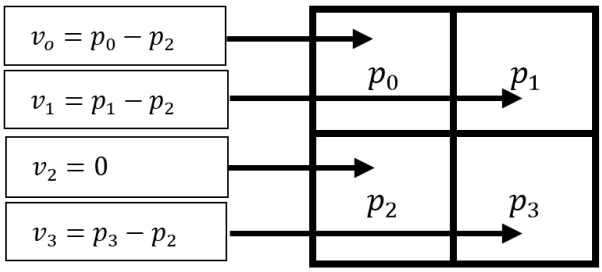

Fig. 3. Calculating the difference between pairs of pixels

$$\left.\begin{aligned} v_0 &= p_0 - p_2 \\ v_1 &= p_1 - p_2 \\ v_2 &= 0 \\ v_3 &= p_3 - p_2 \end{aligned}\right\} \quad (3)$$

Step 3: After the calculation of the difference between pairs of pixel blocks in equation (3) is complete, reduce the difference of (p_0, p_1 and p_3) whose value is greater than 1 ((p_0, p_1 and p_3) > 1) or smaller than -1 ((p_0, p_1 and p_3) < -1). This method is the same as in Al Huti et al. [29], where the value between ($-1 \leq p_n \leq 1$) is not reduced because it can cause distortion. Then RDE is calculated using both parts using (4) before embedding. If $v_n > 1$, the first part is used and if $v_n < -1$, then the second part of the expression is applied.

$$v_n^{"} = \begin{cases} v_n - \left(2^{\lfloor \log_2(v_n) \rfloor} + (2 + \lfloor \log_2(v_n) \rfloor)\right), & if\ b_n > 1 \\ v_n + \left(2^{\lfloor \log_2(v_n) \rfloor} + (2 + \lfloor \log_2(v_n) \rfloor)\right), & if\ v_n < -1 \end{cases} \quad (4)$$

Here, v_n for each block starting from 0 to 3 ($0 \leq v_n \leq 3$), for each block $p_2 = 0$ is not used to hide data. The reduced difference in (4) is different from the previous research [29], shown in (5). As an illustration, calculate the difference using a fixed pixel value as a base point in (3), where the block of pixels $p = (p_0, p_1, p_2, p_3)$ has a pixel value $p_0 = 90$, $p_1 = 75$, $p_2 = 30$ and $p_3 = 55$. By using p_2 as a base point, the difference is calculated using (3), after that, we get a vector p has a difference value p_0, p_1, p_3 as in Fig. 4.

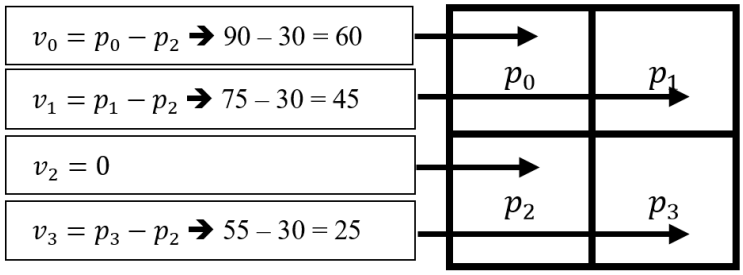

Fig. 4. Illustration of calculating the difference between pixels according to Fig. 3

From the illustration (Fig. 4) as shown the calculation of the difference value between pixels, we find the results are still higher than 1 ((p_1, p_2 and p_3) > 1). The example of the reduction scheme using the method of Al Huti et al. [29] in (5) and the method proposed in (4) displayed in Table 1, and the provided result in Table 2. We find that by using the scheme proposed by (4), the difference values are smaller than those obtained using (5).

$$v_n^{"} = \begin{cases} v_n - \left(2^{\lfloor \log_2(v_n) \rfloor} + \lfloor \log_2(v_n) \rfloor\right), & if\ v_n > 1 \\ v_n + \left(2^{\lfloor \log_2(v_n) \rfloor} + \lfloor \log_2(v_n) \rfloor\right), & if\ v_n < -1 \end{cases} \quad (5)$$

We provide the calculation of new pixels in the stego-image in (6), different from Al Huti et al. [29]. Furthermore, to prevent the cover image from getting worse, the secret data is not embedded in the third pixel of each block (p_2) because this is taken as the base point. To prevent underflow and overflow, each new pixel p'_n in each block must meet the condition of $0 \leq v'_n + p_n \leq 255$ if not all blocks are marked as non-changeable. We use LM to track embedding information on each block.

$$\left.\begin{aligned} p'_0 &= v'_0 + p_2 \\ p'_1 &= v'_1 + p_2 \\ p'_2 &= p_2 \\ p'_3 &= v'_3 + p_2 \end{aligned}\right\} \quad (6)$$

Table 1. Comparison between Al Huti et al. [29] and the proposed methods

Al Huti et al. [29]	Proposed Method
$v_n^{"} - \left(2^{\lfloor \log_2(v_n^{"}) \rfloor} + \lfloor \log_2(v_n^{"}) \rfloor\right)$	$v_n^{"} - \left(2^{\lfloor \log_2(v_n^{"}) \rfloor} + (2 + \lfloor \log_2(v_n^{"}) \rfloor)\right)$
➔ $v_0 = 60$ $v_0^{"} = 60 - \left(2^{\lfloor \log_2(60) \rfloor} + \lfloor \log_2(60) \rfloor\right)$ $v_0^{"} = 60 - (32 + 5) = 23$	➔ $v_0 = 60$ $v_0^{"} = 60 - \left(2^{\lfloor \log_2(60) \rfloor} + (2 + \lfloor \log_2(60) \rfloor)\right)$ $v_0^{"} = 60 - (32 + (2 + 5)) = 21$
➔ $v_1 = 45$ $v_1^{"} = 45 - \left(2^{\lfloor \log_2(60) \rfloor} + \lfloor \log_2(60) \rfloor\right)$ $v_1^{"} = 45 - (32 + 5) = 8$	➔ $v_1 = 45$ $v_1^{"} = 45 - \left(2^{\lfloor \log_2(60) \rfloor} + (2 + \lfloor \log_2(60) \rfloor)\right)$ $v_1^{"} = 45 - (32 + (2 + 5)) = 6$
➔$v_3 = 25$ $v_3^{"} = 25 - \left(2^{\lfloor \log_2(25) \rfloor} + \lfloor \log_2(25) \rfloor\right)$ $v_3^{"} = 25 - (16 + 4) = 5$	➔$v_3 = 25$ $v_3^{"} = 25 - \left(2^{\lfloor \log_2(25) \rfloor} + (2 + \lfloor \log_2(25) \rfloor)\right)$ $v_3^{"} = 25 - (16 + (2 + 4)) = 3$

Table 2. Comparison of values of reduced difference

Value Difference v_n	Reduced Difference	Al Huti et al. [29]	Proposed Method
$v_0 = 60$	$v_0^{"}$	23	21
$v_1 = 45$	$v_1^{"}$	8	6
$v_3 = 25$	$v_3^{"}$	5	3
Average		12	10

On each pixel block in the map, this location is stored in the form of vector. The location map $LM = (LM_1, LM_2, LM_3, LM_4, LM_5)$ is allocated to set bits $1, 0$ and -1 for expandable, changeable and non-changeable block pixels. For the pixel block category can be seen in Fig. 5.

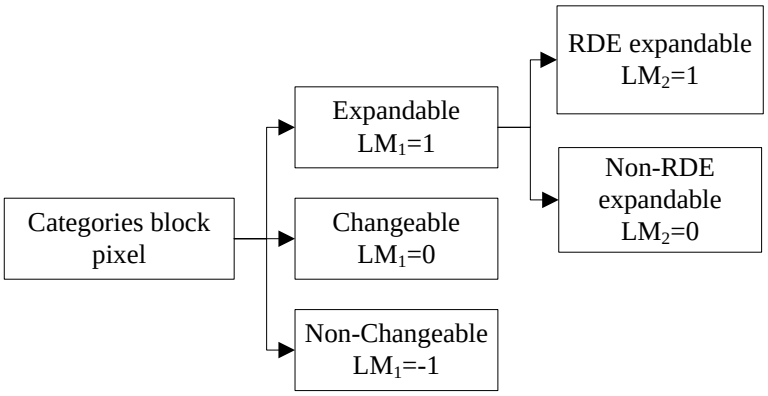

Fig. 5. The structure of the location map

If $v_n = v_n^{"} \pm \left(2^{\lfloor \log_2(v_n^{"}) \rfloor} + (2 + \lfloor \log_2(v_n^{"}) \rfloor)\right)$, then LM_3, LM_4, LM_5 set to 0 and if $v_n \neq v_n^{"} \pm \left(2^{\lfloor \log_2(v_n^{"}) \rfloor} + (2 + \lfloor \log_2(v_n^{"}) \rfloor)\right)$ then the location maps LM_3, $LM_4$and LM_5 take value 1. To distinguish expandable block categories $LM_1 = 1$ and $LM_2 = 0$, then to non-expandable RDE blocks. This non-expandable RDE block is only a value between -1 one directly used without having to be subtracted. Then, LM_3, LM_4, $LM_5 = 0$ is for changeable blocks. If the difference (p_0, p_1 and b_3) is odd, then the location maps LM_3, LM_4, and LM_5 is set to 1 and if the difference has evenly distributed, then this location map is set to 0. Finally, the stego image and location maps still separated. Further, the steps for calculating RDE embedding summarized in Fig. 6.

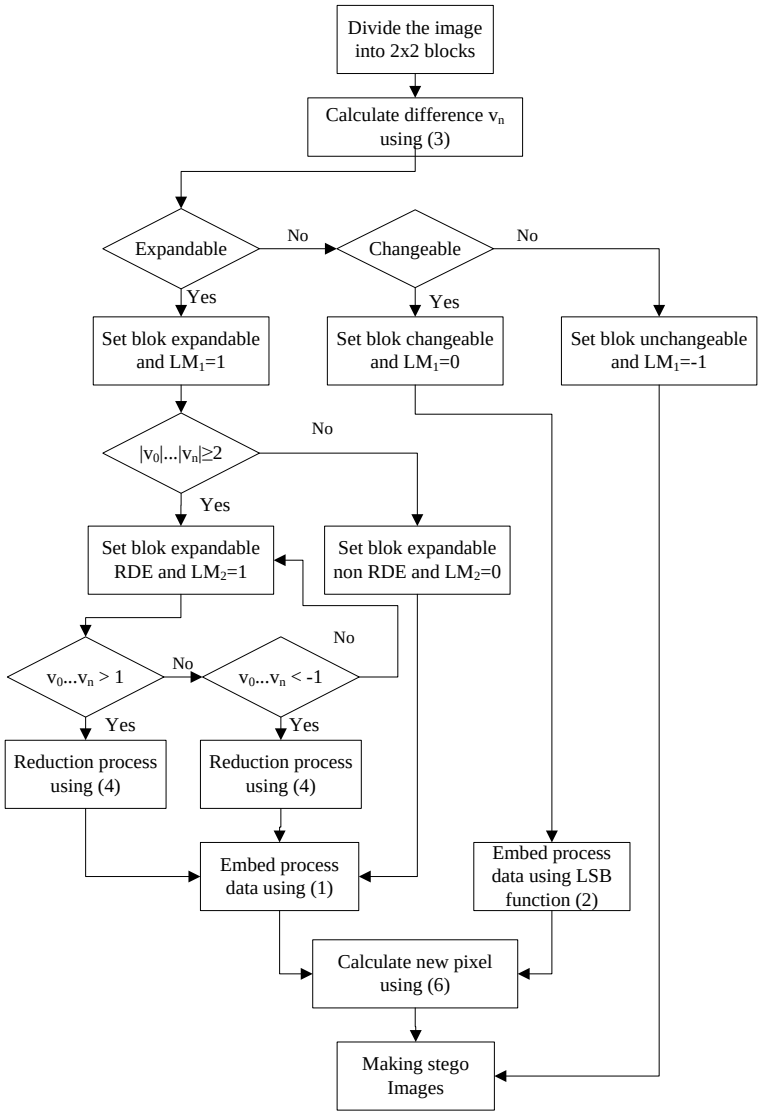

Fig. 6. RDE embedding process

2.2. Data Extraction Stage

The extraction process is done to get the hidden secret message and carried out as follows:

Stage 1: Divide the stego image into blocks; each of them has four pixels. After that, the difference between pixel pairs is calculated using (3), that is, p_n = (p_0, p_1 and p_3) calculated on each subsequent location maps block used to get the secret message and the value of the original pixel. Then do expandable RDE extraction only on location maps $LM_1 = 1$ and $LM_2 = 1$. Non-expandable RDE process if $LM_1 = 1$ and LM_2 = 0, then LM_1 = 0 for blocks changeable can be accessed. To be able to process non-changeable blocks the location map uses $LM_1 = -1$.

Stage 2: Recover original differences and secret bits for expandable RDE. Next look for the secret bits by using LSB extracted from $v_n^{"}$, after which the original differences in the recovered v_n are:

1) If $v_n^{"}$, > 1 and LM_3, LM_4, LM_5 = 0, use equation (7) and if $v_n^{"}$> 1 and LM_3, LM_4, LM_5 = 1, then equation (8) to get the original difference v_n:

$$v_n = v_n^{"} + \left(2^{\lfloor \log_2(v_n^{"}) \rfloor} + (2 + \lfloor \log_2(v_n^{"}) \rfloor)\right) - 1) \quad (7)$$

$$v_n = v_n^{"} - \left(2^{\lfloor \log_2(v_n^{"}) \rfloor} + (2 + \lfloor \log_2(v_n^{"}) \rfloor)\right) - 1) \quad (8)$$

2) If $v_n^{"} < -1$ and LM_3, LM_4, LM_5= 0, use equation (9) and if $v_n^{"} < -1$ and LM_3, LM_4, LM_5 = 1, use equation (10) to get v_n:

$$v_n = v_n^{"} + \left(2^{\lfloor \log_2(v_n^{"}) \rfloor} + (2 + \lfloor \log_2(v_n^{"}) \rfloor)\right) \quad (9)$$

$$v_n = v_n^{"} - \left(2^{\lfloor \log_2(v_n^{"}) \rfloor} + (2 + \lfloor \log_2(v_n^{"}) \rfloor)\right) \quad (10)$$

Stage 3: Calculate the new pixels using equation (11), where $v_n = (v_0, v_1, v_2, v_3)$. To processing non-expandable RDE blocks, the secret message (s) obtained by taking LSB from $v_n^{"}$ in (12) and the expression defined in (13) is used to get v_n.

$$\left.\begin{aligned} p_0 &= v_0 + p_2 \\ p_1 &= v_2 + p_2 \\ p_2 &= p_2 \\ p_3 &= v_3 + p_2 \end{aligned}\right\} \quad (11)$$

$$s = mod \left\lfloor \frac{v_0'}{2} \right\rfloor \quad (12)$$

$$v_n = 2 * \left\lfloor \frac{v_0'}{2} \right\rfloor \quad (13)$$

Then for the changeable block, the secret bit extracts take LSB from $v_n^{"}$ by using the modulus function (mod 2 from $v_n^{"}$), after which the original difference v_n was calculated. If the map LM_3, LM_4, LM_5 = 0 and difference $v_n^{"}$ was odd. Next, the numbers of recovery conducted using (14) and if location maps LM_3, LM_4, LM_5 = 1 and difference $v_n^{"}$ is even, then use (15) to recover v_n.

$$\boldsymbol{v_n = 2 * \left\lfloor \frac{v_0'}{2} \right\rfloor - 1} \quad (14)$$

$$\boldsymbol{v_n = 2 * \left\lfloor \frac{v_0'}{2} \right\rfloor + 1} \quad (15)$$

On the location maps during the extraction process, it is used to track every information about the operations carried out in each block. The steps for RDE extraction displayed in Fig. 7. While the difference between the method in [29] and the proposed method in Table 3.

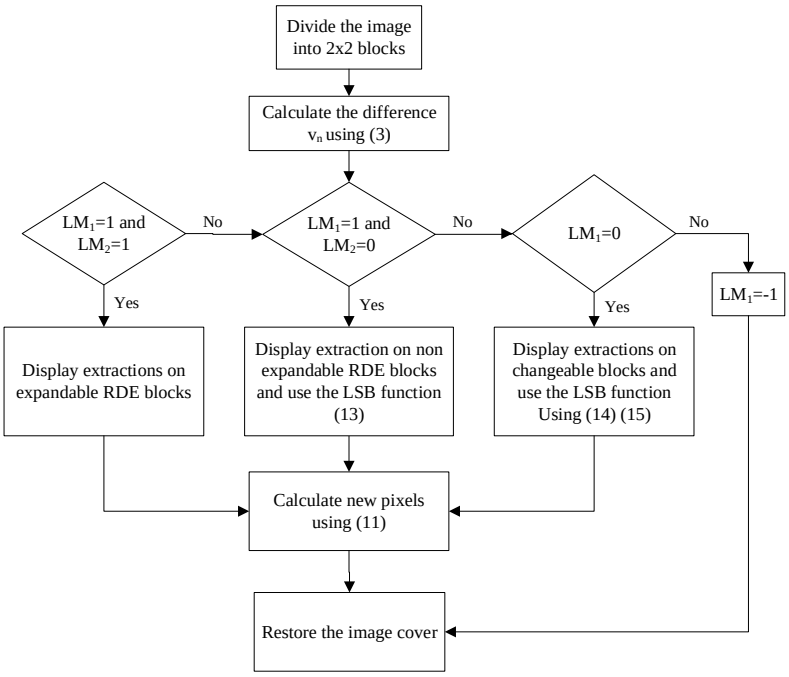

Fig. 7. Process for extraction

Table 3. Comparison between Al Huti et al. [29] and the proposed method

Description	Al Huti et al. [29]	Proposed Method
Pixel block	4 x 4	2 x 2
Base point	v_0	v_2 (constant every block)
Calculate the difference between the pixel pair	$\begin{cases} v_0 = 0 \\ v_1 = p_1 - p_0 \\ v_2 = p_2 - p_1 \\ v_3 = p_3 - p_2 \\ \quad : \quad : \quad : \\ v_{15} = p_{15} - p_{14} \end{cases}$	$\begin{cases} v_0 = p_0 - p_2 \\ v_1 = p_1 - p_2 \\ v_2 = 0 \\ v_3 = p_2 - p_2 \end{cases}$
Reduction function for RDE	$v_n^{"} = \begin{cases} v_n - \left(2^{\lfloor \log_2(v_n) \rfloor} + \lfloor \log_2(v_n) \rfloor\right), & if\ v_n > 1 \\ v_n - \left(2^{\lfloor \log_2(v_n) \rfloor} + \lfloor \log_2(v_n) \rfloor\right), & if\ v_n < -1 \end{cases}$	$v_n^{"} = \begin{cases} v_n - \left(2^{\lfloor \log_2(v_n) \rfloor} + (2 + \lfloor \log_2(v_n) \rfloor)\right), & if\ v_n > 1 \\ v_n - \left(2^{\lfloor \log_2(v_n) \rfloor} + (2 + \lfloor \log_2(v_n) \rfloor)\right), & if\ v_n < -1 \end{cases}$
Calculate new pixels	$\begin{cases} p_0' = p_0 \\ p_1' = v_1^{"} - p_0' \\ p_2' = v_2^{"} - p_1' \\ p_3' = v_3^{"} - p_2' \\ \quad : \quad : \quad : \\ p_{15}' = v_{15}^{"} - p_{14}' \end{cases}$	$\begin{cases} p_0' = v_0' + p_2 \\ p_1' = v_1' + p_2 \\ p_2' = p_2 \\ p_3' = v_3' + p_2 \end{cases}$

3. Results and Discussion

In this research, the extracted secret data used to measure the similarity between the original and the stego images. The purpose of the experimental results is to evaluate the level of distortion from the stego image. In connection with the number of secret bits hidden to measure the capacity, we conducted experiments using MATLAB R2018a that runs on Acer Aspire Laptop V5-471 Series i5-2467M, 500 GB hard disk, 8 GB memory, 1.6 GHz CPU and Windows 10 Professional 64-bit operating system.

This research needs to evaluate how well the proposed method compares with previous research [29]. The images used for evaluation are public standard medical images with 512 x 512 sizes obtained from eMicrobes Digital Library [32], which are Abdominal, Chest, Hand, Head, and Leg (Fig. 8). To find out how far the performance of inserted secret data hides in the image, we use 10 Kb, 20Kb, 50 Kb, 100 Kb, and the maximum capacity.

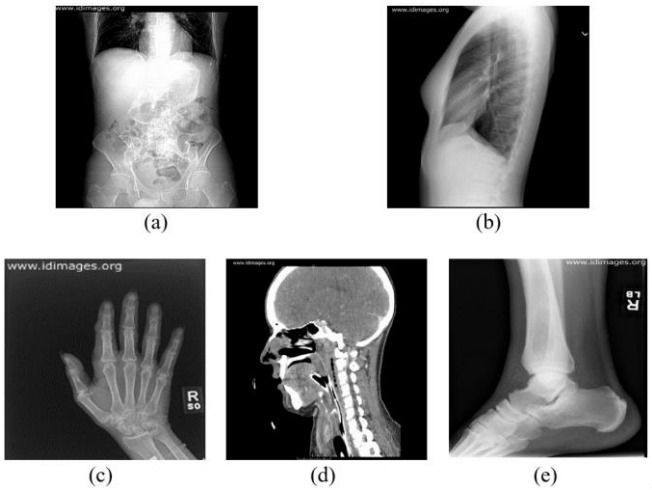

Fig. 8. Medical image [18] (a) Abdominal (b) Chest (c) Hand (d) Head and (e) Leg

In this research, we use Peak Signal to Noise (PSNR) to compare the quality of cover images before and after the secret message is inserted. The value of MSE must be determined first before calculating PSNR. MSE is the average error value between the cover image and the insertion image. The calculation of MSE and PSNR presented in (16) and (17).

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N}\left(S_{ij} - C'_{ij}\right)^2 \quad (16)$$

$$\text{PNSR} = 10 \times \log_{10}\frac{255}{\text{MSE}} \quad (17)$$

Here, i and j are the coordinates of the pixel value of the stego and original images respectively, N is the dimension or size of the image, S_{ij} states stego image and C'_{ij} represents the original image. Then the bits per pixel (bpp) are used to calculate the amount of capacity. Bpp value obtained by dividing the amount of secret data to be inserted and the number of pixels in the original image. In this research, the result of our experiment presented in Table 4 with maximum secret data. Then we use 50 Kb secret data displayed in Table 5. From the experiment, we know that the PSNR is higher than that of the previous method [29].

Table 4. The Comparison between Al Huti et al. [29] and the proposed method by using the maximum capacity

Image Name	Al Huti et al. [29]			Proposed Method		
	Capacity (bit)	*PSNR (dB)*	*Time (s)*	*Capacity (bit)*	*PSNR (dB)*	*Time (s)*
Abdominal	195624	33,421	2.23	193419	34,286	2.22
Chest	196542	38,084	2.55	195834	38,566	2.49
Hand	196227	34,941	2.34	195924	35,170	2.28
Head	194493	31,446	2.10	193596	31,436	2.03
Leg	196389	36,341	2.43	195672	36,767	2.38

Table 5. Comparison between Al Huti et al. [29] and the proposed method using 50 Kb of secret data

Image Name	Al Huti et al. [29]			Proposed Method		
	Capacity (bit)	*PSNR (dB)*	*Time (s)*	*Capacity (bit)*	*PSNR (dB)*	*Time (s)*
Abdominal	49674	35,582	0.78	47607	37,440	0.77
Chest	49875	39,534	0.87	49464	40,524	0.83
Hand	49680	37,896	0.83	49494	38,507	0.79
Head	49593	38,157	0.84	49110	39,108	0.81
Leg	49743	38,112	0.84	49443	38,614	0.80

Fig. 9 illustrates the original image after hiding secret data obtained stego image. We can see that the visual quality of the cover image can be maintained. Besides, the original image and the stego image presented in Fig. 9(a) and 9(b) as well as 9(c) and 9(d) where the two images are almost similar and relatively difficult to identify the differences between them (original image and image stego). From the two images, there is a high similarity between the stego image and the original cover image maintained an excellent quality stego image.

The results of the visualization of the performance of the scheme in Al Huti et al. [29] and the proposed one presented in Table 4 and Table 5. We can see that after hiding the secret message, the proposed method outperformed the scheme of Al Huti et al. [29] in terms of visual quality. The average PSNR is calculated based on the size of the secret message used to evaluate the level of distortion (or change) found on the cover image after hiding each message size. For example for the cover image of Abdominal, Chest, Hand, Head and Leg, the average PSNR obtained by calculating the maximum secret PSNR data (PSNR from each cover image. That is, Abdominal =34,286 dB, Chest =38,566 dB, Hand =35,170 dB, Head =31,436 dB, and Leg =36,767 dB). Then, for the average PSNR using 50 Kb of secret data (PSNR from each cover image that is Abdominal =37,440 dB, Chest =40,524 dB, Hand =38,507 dB, Head =39, 108 dB and Leg =38,614 dB). This research used medical imagery because it has very high redundancy and hidden without much distorting the image. The proposed approach not

only improves the quality but also the number of bits that can be hidden in the image. In general, the proposed approach can be suitable for all users depending on the embedding capacity needed.

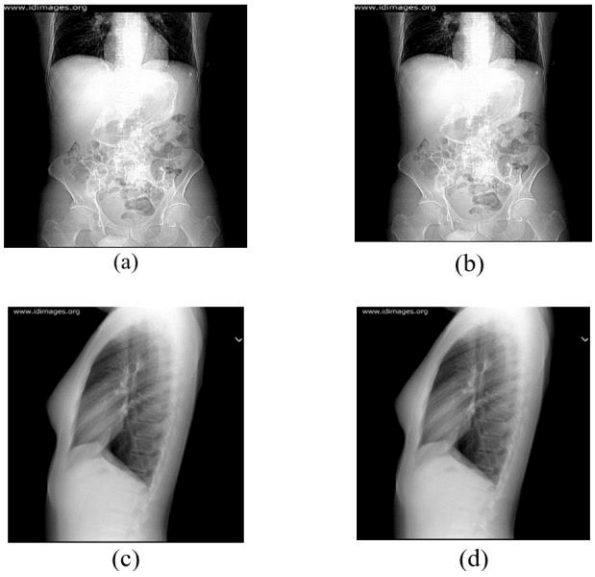

Fig. 9. An example medical image (a) Abdominal image before hiding data (b) Abdominal image after Hiding 50 Kb (c) Chest image before hiding data (d) Chest image after hiding 50 Kb

4. Conclusion

This research presents the development of reversible hiding data with modifications to reduce different expansion. In this proposed method, we have improved the performance of embedding the secret data in an image. For this purpose, we vary the size of the pixel blocks, determine the base point, and improve the reduction scheme. It is designed to refine existing technical abilities, especially those presented by previous researchers. The experimental results show that the proposed method can increase the PSNR value. In future work, we will focus on increasing the embedding capacity and conserving the quality of the cover image while evaluating the proposed algorithm.

Acknowledgment

The authors would like to thank you both Institut Teknologi Sepuluh Nopember (ITS) and Sekolah Tinggi Teknologi Angkatan Laut (STTAL) for the support.

References

[1] C. Renato and N. María, "Technologies' Application, Rules, and Challenges of Information Security on Information and Communication Technologies," *Proc. - 2015 Asia-Pacific Conf. Comput. Syst. Eng. APCASE 2015*, pp. 380–386, 2015, doi: 10.1109/APCASE.2015.74.

[2] X. Zhang, J. Wang, Z. Wang, and H. Cheng, "Lossless and Reversible Data Hiding in Encrypted Images with Public Key Cryptography," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 9, pp. 1622–1631, 2015, doi: 10.1109/TCSVT.2015.2433194.

[3] W. Zhang, K. Ma, and N. Yu, "Reversibility improved data hiding in encrypted images," *Signal Processing*, vol. 94, pp. 118–127, 2014, doi: 10.1016/j.sigpro.2013.06.023.

[4] K. S. Seethalakshmi, Usha B A, and Sangeetha K N, "Security enhancement in image steganography using neural networks and visual cryptography," in *2016 International Conference on Computation System and Information Technology for Sustainable Solutions (CSITSS)*, 2016, pp. 396–403, doi: 0.1109/CSITSS.2016.7779393.

[5] M. Hussain, A. W. A. Wahab, Y. I. Bin Idris, A. T. S. Ho, and K. H. Jung, "Image steganography in spatial domain: A survey," *Signal Process. Image Commun.*, vol. 65, no. March, pp. 46–66, 2018, doi: 10.1016/j.image.2018.03.012.

[6] M. A. P. T. Dileep, Alka, K.Anusudha, "An Efficient Reversible Data Hiding Technique In Encrypted Images Based On Chaotic Map," *Int. Conf. Control. Commun. Comput. Technol.*, pp. 539–543, 2015, doi: 10.1109/ICCICCT.2015.7475338.

[7] W. Bender, D. Gruhl, N. Morimoto, and A. Lu, "Techniques for data hiding," *IBM Syst. J.*, vol. 35, no. 3.4, pp. 313–336, 1996, doi: 10.1147/sj.353.0313.

[8] Y. Tsai, D. Tsai, and C. Liu, "Reversible data hiding scheme based on neighboring pixel differences," *Digit. Signal Process.*, vol. 23, no. 3, pp. 919–927, 2013, doi: 10.1016/j.dsp.2012.12.014.

[9] S. Weng, Y. Liu, J. Pan, and N. Cai, "Reversible data hiding based on flexible block-partition and adaptive block-modification strategy," *J. Vis. Commun. Image Represent.*, 2016, doi: 10.1016/j.jvcir.2016.09.016.

[10] D. Lou, M. Hu, and J. Liu, "Multiple layer data hiding scheme for medical images," *Comput. Stand. Interfaces*, vol. 31, pp. 329–335, 2009, doi: 10.1016/j.csi.2008.05.009.

[11] G. Gao, X. Wan, S. Yao, Z. Cui, C. Zhou, and X. Sun, "Reversible data hiding with contrast enhancement and tamper localization for medical images," *Inf. Sci. (Ny).*, vol. 386, pp. 250–265, 2017, doi: 10.1016/j.ins.2017.01.009.

[12] T. Lu, C. Tseng, and J. Wu, "Dual imaging-based reversible hiding technique using LSB matching," *Signal Processing*, vol. 108, pp. 77–89, 2015, doi: 10.1016/j.sigpro.2014.08.022.

[13] M. Tayel, A. Gamal, and H. Shawky, "A Proposed Implementation Method of an Audio Steganography Technique," *2016 18th Int. Conf. Adv. Commun. Technol.*, no. 3, pp. 180–184, 2016, doi: 10.1109/ICACT.2016.7423320.s

[14] J. M. Blackledge and A. I. Al-Rawi, "Steganography Using Stochastic Diffusion for the Covert Communication of Digital Images," *IAENG Int. J. Appl. Math.*, vol. 41, pp. 270–298, 2011, available at: Google Scholar.

[15] H. S. E. S. F. El-zoghdy and O. S. Faragallah, "Adaptive Difference Expansion-Based Reversible Data Hiding Scheme for Digital Images," *Arab. J. Sci. Eng.*, vol. 41, pp. 1091–1107, 2016, doi: 10.1007/s13369-015-1956-7.

[16] M. S. Subhedar and V. H. Mankar, "Current status and key issues in image steganography: A survey," *Comput. Sci. Rev.*, pp. 1–19, 2014, doi: 10.1016/j.cosrev.2014.09.001.

[17] E. Ghasemi, J. Shanbehzadeh, and N. Fassihi, "High Capacity Image Steganography using Wavelet Transform and Genetic Algorithm," *Proc. Int. MultiConference Eng. Comput. Sci.*, vol. I, 2011, doi: 10.1007/978-1-4614-1695-1_30.

[18] R. Tanwar and M. Bisla, "Audio Steganography," *2014 Int. Conf. Optim. Reliab. Inf. Technol.*, pp. 322–325, 2014, doi: 10.1109/ICROIT.2014.6798347.

[19] D. C. Kar and C. J. Mulkey, "A multi-threshold based audio steganography scheme," *J. Inf. Secure. Appl.*, vol. 23, pp. 54–67, 2015, doi: 10.1016/j.jisa.2015.02.001.

[20] M. B. Andra, T. Ahmad, and T. Usagawa, "Medical Record Protection with Improved GRDE Data Hiding Method on Audio Files," *Eng. Lett.*, vol. 25, pp. 112–124, 2017, available at: Google Scholar.

[21] J. Tian, "Reversible Data Embedding Using a Difference Expansion," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 8, pp. 890–896, 2003, doi: 10.1109/TCSVT.2003.815962.

[22] A. M. Alattar, "Reversible watermark using difference expansion of quads," *IEEE Trans. Image Process.*, vol. 13, no. 1, pp. 377–380, 2004, doi: 10.1109/TIP.2004.828418.

[23] A. M. Alattar, "Reversible watermark using the difference expansion of a generalized integer transform," *IEEE Trans. Image Process.*, vol. 13, no. 8, pp. 1147–1156, 2004, doi: 10.1109/TIP.2004.828418.

[24] D. M. Thodi and J. J. Rodríguez, "Expansion embedding techniques for reversible watermarking," *IEEE Trans. Image Process.*, vol. 16, no. 3, pp. 721–730, 2007, doi: 10.1109/TIP.2006.891046.

[25] C. L. Liu, D. C. Lou, and C. C. Lee, "Reversible data embedding using reduced difference expansion," in *Proceedings - 3rd International Conference on Intelligent Information Hiding and Multimedia Signal Processing, IIHMSP 2007.*, 2007, vol. 1, pp. 433–436, doi: 10.1109/IIH-MSP.2007.267.

[26] H. Yi, S. Wei, and H. Jianjun, "Improved reduced difference expansion based reversible data hiding scheme for digital images," in *ICEMI 2009 - Proceedings of 9th International Conference on Electronic Measurement and Instruments*, 2009, pp. 4315–4318, doi: 10.1109/ICEMI.2009.5274054.

[27] T. Ahmad, M. Holil, W. Wibisono, and I. Royyana Muslim, "An improved Quad and RDE-based medical data hiding method," in *Proceeding - IEEE CYBERNETICSCOM 2013: IEEE International Conference on Computational Intelligence and Cybernetics*, 2013, pp. 141–145, doi: 10.1109/CyberneticsCom.2013.6865798.

[28] Y. Kurniawan, L. A. Rahmania, T. Ahmad, W. Wibisono, and R. M. Ijtihadie, "Hiding secret data by using modulo function in quad difference expansion," in *2016 International Conference on Advanced Computer Science and Information Systems, ICACSIS 2016*, 2017, pp. 433–438, doi: 10.1109/ICACSIS.2016.7872741.

[29] M. H. A. Al Huti, T. Ahmad, and S. Djanali, "Increasing the capacity of the secret data using DEpixels blocks and adjusted RDE-based on grayscale images," in *Proceedings of 2015 International Conference on Information and Communication Technology and Systems, ICTS 2015*, 2016, pp. 225–230, doi: 10.1109/ICTS.2015.7379903.

[30] J. Y. Hsiao, K. F. Chan, and J. Morris Chang, "Block-based reversible data embedding," *Signal Processing*, vol. 89, no. 4, pp. 556–569, 2009, doi: 10.1016/j.sigpro.2008.10.018.

[31] T. D. Kieu and C. C. Chang, "A high stego-image quality steganographic scheme with reversibility and high payload using multiple embedding strategy," *J. Syst. Softw.*, vol. 82, no. 10, pp. 1743–1752, 2009, doi: 10.1016/j.jss.2009.05.028.

[32] eMicrobes Digital Library, "Partners Infectious Disease Images", Accessed 30 March 2019, available at: http://www.idimages.org/images/browse/ImageTechnique/.